Tle

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/752,541	12/29/2000	Stephen Boyd	4363P001	1435

7590 09/21/2006

Daniel M. DeVos
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
Seventh Floor
12400 Wilshire Boulevard
Los Angeles, CA 90025-1026

EXAMINER	
VU, TUAN A	

ART UNIT	PAPER NUMBER
2193	

DATE MAILED: 09/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/752,541	BOYD ET AL.
	Examiner	Art Unit
	Tuan A. Vu	2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *08 September 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *5-11 and 23-52* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *5-11 and 23-52* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the application filed 9/8/06. As indicated by Applicants, no claims have been amended. Claims 5-11, 23-52 are re-submitted for examination.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claims 5-9, 23-28, 29-37, 38-43, and 44-52 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a " useful, concrete, and tangible result" be accomplished. An "abstract idea" when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the "useful arts" when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a "useful, concrete and tangible result".

Claims 5 and 23 recite reading from a source file; converting algebraic expressions therein into signomial expressions; and reducing these into some other forms for the process of preparing input for a geometric program solver. As perceived from claim 5 and 23, the computer method claimed is to prepare input expressions for a process (program solver) which is not described; hence the claimed subject matter amounts to having some algebraic expressions being converted into reduced forms of signomial expressions to serve as input. In the hypothetical case where the reduced expressions are the end results, the claim still lacks reasonable description to convey a tangible form of such results, i.e. what tangible form the final reduced expressions would have been stored in or represented as (e.g. a file, computer-medium or a persisted set of records or mass storage component, as opposed to mere expressions construed from a mental

derivation process, thus not real world entities). It can be interpreted from the claim as it is that the expressions have been computer-generated after some conversion; and converted data inside the confines of a computer remain internal thereto and stays as digital/abstracted entities (regardless as to whether or not they are considered final or intermediate product) when they are expected to be tangible in terms of being of some use to the user who executes the computer instructions leading to such transformed data. That is, absent any tangible and concrete embodiment to store the result yielded (e.g. is it a executable file or metadata/library file or DB record persisted in a file system?); or absent any steps for teaching a proper use of the reduced expressions thus recited in terms of user/human interface or machine-operated interaction (Application level) so to lead to a useful result, the above claims remain an non-practical application or an abstract functionality (e.g. a reduced algebraic format that stays conceptual without being materialize into a real world data) and are rejected for leading to a non-statutory subject matter. Hence claims 6-11, 24-28 are also rejected for leading to a non-statutory matter for failing to remedy to the deficiency of the base claims.

Claim 29 recites the same subject matter of claim 23 hence is rejected for the same reasons. Claims 30-37 fail to remedy the deficiencies of claim 29 hence are rejected for the same reasons as the base claim.

Claims 38 and 44 recite similar limitations as claim 23, 29 hence are rejected for also leading to a non-statutory subject matter. Claims 39-43, and 45-52 are also rejected for not remedying to the deficiencies of the base claims from which they depend.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 5-11, 23-52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shao-Po et al., "A Parser/Solver for Semidefinite Programs with Matrix Structure", Technical Report, Information System Laboratory, Stanford University, November 1995 (hereinafter Shao-Po – provided in IDS), in view of Hershenson et al., USPN: 6,311,145 (hereinafter Hershenson), and further in view of Dennis Bricker, "Signomial Geometric Programming", University of Iowa, April 1999, http://css.engineering.uiowa.edu/~dbricker/Stacks_pdf8/Signomial_GP.pdf (hereinafter Bricker).

As per claim 5, Shao-Po discloses a computer-implemented method of parsing a mathematical optimization problem comprising:

creating a set of mathematical expressions or constraints from the mathematical terms(e.g. *equality constraints* - ch. 4.3.1 pg. 85; *Lyapunov inequality* - ch. 4.4.1, pg. 86);

converting said set of constraints expressions to a optimized compact numeric format to be accepted by a program solver (e.g. *matrix* 4.14 – pg. 87; *spdsol* language & equ. 4.15 – pg. 89 – Note: compacting separate math expression into one matrix format reads on compact numeric format).

Shao-Po does not specifically disclose reading from a source file when mentioning about reading the matrix specifications or the math problem specifications or matrix data structure; but reading discloses Matlab facilities and sdpsol language input for a parser and containing matrix

variables reads on user's defined Matlab program or source file for defining those variables – see ch. 4.2.1, pg. 82) and the parser receiving input having a plurality of algebraic expressions that represent a mathematical problem, each algebraic expression in said plurality having one or more mathematical terms (e.g. *minimize* -, eq. 4.1, pg. 79; eq. 4.2 pg. 80; eq. 4.3 – pg. 81). Based on the Matlab facilities to help developers to specify a problem using math representation and the input files well known in Matlab, the sdpsol parser reading an input file would have been strongly suggested for such source file specifying the developers math or matrix structure specification to be an obvious feature. One skill in the art would be motivated to user Matlab parser and submit to Shao-Po's parser with the problem specifications (matrix structure variables) in a form of a source file for the Matlab parser to operate, because file are easy to maintain or utilized as a means to persist and transport developer's program specifications related to a problem, such file having the language as taught by Shao-Po sdpsol language and using its parser as taught by Matlab.

But Shao-Po does not specify that the mathematical terms or constraints are converted into a set of signomial expressions; nor does Shao-Po explicitly specify converting those set of signomial expressions into a format accepted by the geometric program solver. However, Shao-Po discloses parser/solver using software *MatLab*, *Bison* and *Flex* (see ch. 4.3 – pg. 84); hence has disclosed submission of matrix-implemented/geometric constraints into a computer-based geometric program solver. Shao-Po further discloses an implemented method for optimizing of circuit design (ch. 4.1, pg. 79; Fig. 4.1 – pg. 87; Fig. 4.4, pg. 90) using a *sdpsol* language programming (e.g. ch. 4.2.3 pg. 83-84. The matrix-implemented/geometric expressions using objectives and constraints are indicative of, or implicitly disclose that a form of signomials; and

in case Shao-Po does not already teach that those constraints are signomial forms, such feature would have been obvious.

Hershenson, in a analogous method to Shao-Po's to optimize a circuit design lumping parametric constraints into a specific set of expressions, discloses optimizing complex non-linear problems (e.g. induction or RF mathematics are non-linear as in Shao-Po's differential equation applying Lyapunov's case) and expressing the constraints or inequalities into posynomials and submitting these to solver using a geometric programming language (e.g. Fig. 1; col. 5, line 34 to col. 10, line 45). It would have been obvious for one of ordinary skill in the art at the time the invention was made to implement the constraints as taught by Shao-Po into sygnomial expressions (Note: posynomial is interpreted as poly form of single sygnomial) as taught by Hershenson if the resources are such that Hershenson's geometric programming language can be implemented because this widely known programming technology operating upon convex or non-linear complex functions can be an efficient tool for effecting improved algorithms to solve problems like those non-linear complex inductive circuitry; and optimizing circuit designs as mentioned by Shao-Po, by solving constraints formed as posynomials, or set of sygnomials as claimed (see Hershenson, col. 1, 2).

Nor does Shao-Po disclose that at least one of the signomial expressions has a negative coefficient. Solving non-definite and complex problems such as Shao-Po's method implies dealing with complex, imaginary numbers or floating points and real numbers; and implementing geometric programming with signomials similar to Shao-Po, with such signomials handling non only positive coefficients but also negatively signed coefficients is evidenced in Bricker's approach (Signomial function, *sign of coefficient* - pg. 1). It would have been obvious for one

of ordinary skill in the art at the time the invention was made to implement the constraints as taught by Shao-Po' matrix posynomials so that the coefficients can be negative as taught by Bricker because of the increased power in dealing with more complex situations and enhance the range of coefficient to address both negative and positive domain of the signomials as shown by Bricker.

As per claim 6, Shao-Po discloses an objective (eq. 4.1- pg. 79) and a set of constraints (e.g. *constraint lyap, constraint equ* – ch. 4.2.2 pg. 84).

As per claim 7, Shao-Po discloses one or more mathematical expressions (e.g. ch. 4.1, pg. 79; Fig. 4.1 – pg. 87; Fig. 4.4, pg. 90) and inequality (e.g. *Lyapunov inequality* – ch. 4.4.1, pg. 86).

As per claim 8, Shao-Po discloses optimization variables (matrices, vector – ch. 4.2.3 – pg. 83- Note: matrix or structures used for the optimization process are optimization variables)

As per claim 9, Shao-Po discloses before converting determining that the mathematical expressions reduce to objective or inequality or equality (e.g. ch. 4.2.2-4.2.3 pg. 83-84); but does not specify reducing expressions into posynomial expressions or determining that such optimization problem is a geometric program. This limitation, however, would have been obvious in view of the rationale set forth in claim 5 using Hershenson's teachings.

As per claim 10, only Hershenson discloses that some expressions are not posynomial expressions (col. 7, line 56 to col. 8, line 27). In light of the rationale set forth in claim 5, it would have been obvious for one of ordinary skill in the art at the time the invention was made to implement the step of determining which expressions are not fit to be further converted into posynomial form as taught by Hershenson and apply such determination step to the problem

solving using constraints-based optimization language by Shao-Po; because if the purpose is to convert complex functions constraints and parameters into posynomial forms, it is required to only focus on creating posynomial expressions and filter out non-posynomial expressions in order to conform to the geometric programming as suggested by Hershenson.

But neither Shao-Po nor Hershenson discloses reporting to a user which expressions cannot be reduced into a posynomial objective or equality/inequality. The implemention of user interface in computer-implemented hardware/software design or circuit emulation framework in order to allow user to author or specify requirements and receive feedback from constraints compatibility checking was a known concept in the programming art at the time the invention was made, especially when such design involve CAD tools as suggested by Hershenson (col. 1, 2) or *LMITool* by Shao-Po (e.g. ch. 4.1.4 - pg. 82). It would have been obvious for one of ordinary skill in the art at the time the invention was made to add to the combination of Hershenson/Shao-Po an user interface allowing the user to interact with the circuit design and algorithmic programming as suggested by Hershenson; as well as the reporting to the users to the effect that some expressions fail to be reduced into posynomial objective or equality/inequality as claimed above. The motivation is that this would allow the user to specify and learn upon the results of such requirement acceptance by the framework or optimization of parameters used in implementing the functions of the circuitry, as applied by common known methodologies like HDL, Verilog-based hardware/software circuit designs.

As per claim 11, the reduction of simple monomial expressions into more posynomial has been taught and addressed in claim 5 (see Hershenson: col. 5, line 34 to col. 10, line 45-- Note: the monomial expressions representing signal mathematics in a circuitry used to be

converted in more complex posynomial are mathematical expressions expressing signals, hence signomial); but Hershenson does not explicitly specifying canceling a combination of signomials. Official notice is taken that simplication of mathematical expressions prior to submitting them to more complex integrations was a known concept at the time the invention was made. Hence it would have been obvious for one of ordinary skill in the art at the time the invention was made to provide the simplification by canceling out signomial combinations in view of the in both optimization methods by Hershenson or Shao-Po, and apply such canceling to Hershenson's method as it enhances the optimization method by Shao-Po as set forth in claim 5 because simplifying a mathematical expression or in this case signomial combination is a must-do step in computation lest extraneous data complications and resources wasting down the later computing stages occur.

As per claim 23, Shao-Po discloses computer-implemented method, comprising preparing input expressions for a geometric program solver (e.g. . *matrix* 4.14 – pg. 87; *spdsol* language & equ. 4.15 – pg. 89) by executing:

converting the plurality of algebraic expressions that represent a geometric program (e.g. *parser*, *MatLab*, *Bison*, *Flex* -- see ch. 4.3 – pg. 84), said converting comprising for each algebraic expression of said plurality of algebraic expressions (e.g. *matrix* 4.14 – pg. 87; *spdsol* language & equ. 4.15 – pg. 89 – Note: compacting separate math expression into one matrix format reads representing algebraic expressions of a geometric program into an accepted format for the program solver):

converting said algebraic expression into a matrix-implemented/geometric expression by converting terms of said matrix-implemented expression into a matrix-implemented/geometric

function (ch. 4.2.2-4.2.3 pg. 83-84; ch. 4.3, 4.4, pg. 84-88 – Note: example 4.13 by Lyapunov reads on function converted from a generic geometric expression to perform a real-life function).

The limitation about reading from a source file has been addressed in claim 5.

Shao-Po does not explicitly disclose that the matrix-implemented/geometric expression is a signomial; nor does Shao-Po disclose reducing said signomial expression to one of the following: a posynomial objective, a posynomial inequality, a monomial equality; but the rationale as to why the signomial limitation would have been obvious has been set forth in claim 5. Further, Shao-Po discloses objective, constraints including inequalities (e.g. ch. 4.2.2-4.2.3 pg. 83-84), hence the posynomial objective, a posynomial inequality, a monomial equality limitations would also have been obvious in light of the signomial limitation being obvious as set forth in claim 5.

As per claim 24, Shao-Po discloses making a substitution in an expression that has an internal variable that represents a previously assigned expression (e.g. *support, assigned internal variables ... later be used* --pg. 82, ch. 4.2.1 – Note: parser to support internal variable definitions reads on substitution for implementing program solving expressions).

As per claim 25, Shao-Po teaches algebraic manipulation (re claim 5) but Shao-Po does not explicitly disclose simplifying the signomial expression by canceling two identical signomial functions of opposite sign; but the concept of canceling two entities with opposite sign is an implicit and basic teaching in all mathematical operations; hence this is implicitly disclosed.

As per claim 26, Shao-Po discloses finding said algebraic expressions within lines of an input source file (e.g. *support, assigned internal variables ... later be used* --pg. 82, ch. 4.2.1 – Note: Matlab like grammar reads on lines of source code being parsed based on grammar rules).

As per claim 27, Shao-Po (combined with Hershenson/Brisker) discloses each one of said algebraic expressions is one of the following: an objective function; an equality constraint, an inequality constraint (refer to claim 23).

As per claim 28, Shao-Po (combined with Hershenson/Brisker) discloses that said geometric program is a signomial program (ch. 4.2.2-4.2.3 pg. 83-84 and the teachings by Hershenson and Brisker as set forth in claims 23, 5).

As per claim 29, Shao-Po discloses a computer implemented method, comprising preparing input expressions for a geometric program solver (e.g. . *matrix* 4.14 – pg. 87; *spdsol* language & equ. 4.15 – pg. 89) by executing:

converting the plurality of algebraic expressions that represent a geometric program (e.g. *parser*, *MatLab*, *Bison*, *Flex* -- see ch. 4.3 – pg. 84), said converting comprising for each algebraic expression of said plurality of algebraic expressions (e.g. *minimize* -, eq. 4.1, pg. 79; eq. 4.2 pg. 80; eq. 4.3 – pg. 81);

combining mathematical terms of said algebraic expression to reduce said algebraic expression to one of the following: a objective, an inequality, an equality (e.g. ch. 4.2.2-4.2.3 pg. 83-84).

The limitation about reading from a source file has been addressed in claim 5.

Shao-Po does not disclose that the reduced form of objective, inequality, equality are respectively a posynomial objective, a posynomial inequality, a monomial equality; but in view of the similarity of matrix-implemented implementing of geometric problems of Shao-Po and posinomials by Hershenson, these limitations would have been obvious as set forth in claims 5 and 23.

As per claims 30 and 31, these claims refer to mathematical terms identifying one of the group of signomial, posinomial, monomial; but since these forms have been addressed in claim 29; these limitations would have been obvious also owing to the implicit teaching that any mathematical polynomial can be either poly/mono-mial and to the posinomials/signomial teachings from the combination Shao-Po, Hershenson and Birsker as set forth in claim 5.

As per claim 32, Shao-Po does not explicitly disclose that said combining mathematical terms comprises determining if operators and functions that relate said mathematical terms permit said reduction. Official notice is taken that reduction of mathematical expression being based on operator (e.g. division, left and right side of equality, multiplication by same number) and type of expressions (common denominator/factor, most common divisor, parentheses… etc) under which those math terms are formed. Thus, based on such well-known, the reduction provided via Software like Mathlab, for example, as taught by Shao-Po implicitly disclose determining if operators and functions that relate said mathematical terms permit said reduction.

As per claims 33 and 34, Shao-Po does not explicitly disclose that said posynomial inequality is a posynomial function less than one and said monomial inequality is a monomial function equal to one. But the chance that a polynomial or posinomial or a monomial be less than one or equal to one is not excluded from all the possible values taken from resolving the matrix-implemented geometric expressions as taught by Shao-Po (ch. 4.3, or 4.4, pg. 84-91). Hence, Shao-Po has disclosed the limitations of claims 33 and 34.

As per claims 35-37, these claims are rejected with the same rationale as set forth in claims 26-28, respectively.

As per claim 38, Shao-Po discloses a method comprising preparing input expressions for a geometric program solver (e.g. . *matrix* 4.14 – pg. 87; *spdsol* language & equ. 4.15 – pg. 89) by executing:

converting (the plurality of algebraic expressions that represent a geometric program…)

converting (into a … expression by converting terms … into a …); all of which steps being the same as recited in claim 23.

These limitations thus are rejected using the corresponding rejection as set forth therein.

The limitation about reading from a source file has been addressed in claim 5.

But Shao-Po does not disclose program code embedded on a readable medium which when executed by a computer causes a method to perform the above steps limitations. But the providing of software embodied in a readable medium for solving a problem would have been obvious in today's selling of product using Matlab or other parsing tools such as taught by Shao-Po.

As per claims 39-43, these claims correspond to claims 24-28; and are rejected with the same rationale as set forth in claims 26-28, respectively.

As per claim 44, this claim corresponds to claim 29; and is rejected with the corresponding rejection as set forth therein, and further includes the computer–readable medium as addressed in claim 38.

As per claims 45-52, these claims are rejected with the same rationale as set forth in claims 30-37, respectively.

Response to Arguments

6. Applicant's arguments filed 9/08/2006 have been fully considered but they are not persuasive. Following are the Examiner's observations in regard thereto.

35 USC §101 Rejection:

(A) Applicants have submitted that providing a format accepted by a computer-aided program solver would be sufficient to overcome the rejection and because action taken is performed by a computer (see Appl. Rmrks, pg. 13, top para). The basis of a method claim is to analyze the claim as a whole in order to perceive whether or not the steps performed would lead to a transformation of data being actually realized so that the result thus realized would be tangible, concrete and useful in the context/onset as set forth in the of the claimed invention. As interpreted, it is unclear as to whether the step of reducing performed by a computer lead to an end result that can be reasonably construed as been collected in tangible form. For example, the recited 'compact numeric format' is seen as intended for being accepted by a program solver, which make the final result a format of unknown nature to be used later by some solving engine. Computer data purported as input into some other solver (most likely some computer process) will be treated as abstract data: the reduced expressions being submitted as input amount to but abstract concept (or software or digitized data) being submitted to a possibly software tool (i.e. a program solver); and the action ends there. It is even more obscure as to whether a plurality of compacted format coming from a plurality of signomial expressions amounts to a readable file, or a executable block of code, an archive, or intermediate stream, or mere mathematical derivations of abstracted nature. There would be no way of telling if the expressions such as those reduced expressions are stored as tangible form to be meaningfully retrieved or used by a

user trying to operate the (inferred) computer-implemented solver. The rejection is therefore maintained.

35 USC §103 Rejection:

(B) Applicants have submitted that it is unclear to Applicants as to whether Examiner is citing Shao-Po reference to meet the recited features of claim 5, or taking Official notice there against (Appl. Rmrks, pg. 15, 1st para). This rationale as to why signomial programs limitation would have been obvious has used the Shao-Po's working on a language and the specifications for representing a problem with matrix; and along with Shao-Po's use of Matlab and in view of complex math involved, has combined teaching from Shao-Po with similar endeavors from Hershenson working with complex math polynomials and from there to Bricker; and no Official Notice is inferred or explicitly applied.

(C) Applicants have submitted that the Office Action contending that Shao-Po's dealing with semidefinite Programming or max-det Problems cannot extend Shao-Po's field of analysis and problem solving to that under geometric programs realm as contemplated by the present Invention (Appl. Rmrks,2nd para - pg. 15). There is no deliberate and exclusive definition of what is referred to a GP in the Specifications apart from the Background section wherein GP amounts to optimization problem having constraints that must meet math criteria, such that GP can be used to describe a circuit design (emphasis added) problem (see BACKGROUND). It is therefore noted that since there is no clear cut definition of GP for it to impart specific difference from what is construed from reading the claim, the signomial expression amount to expression with complex sequence of polynomials with coefficient of complex constructs, and this is perceived via circuit design of Shao-Po – its form of polynomial complexity; which is furthered

by evidences in Hershenson's *posynomials* to represent such complex designs; and this is what the Background is referring to. Moreover, the claim does not establish sufficient teachings as to preclude Shao-Po math or circuit design expressions from being expressed also as signomial type of for representing complex circuit design equations of the form taught in Hershenson's posynomials. That is, there is no clear definition in the part of the claim or the Specifications that would reasonably preclude the equation-based approach by Shao-Po's problem solving or circuit design analysis from using a polynomial forms like those exemplified by Hershenson, i.e. being a signomial; simply because what is claimed is not *GP* but *signomial expression*. Applicants fail to show accurate and factual demonstration as to why the above combination would have been non-obvious or would lead to adverse effect. Applicants argue that semidefinite programs endeavor cannot be extended to GP; when there is no specific in the claims or even in the Specifications as to (i) support why Applicant suddenly equate 'signomial expressions' and 'GP' as one entity; and as (ii) to differentiate a signomial expressions from the combined circuit design polynomial representations by both Shao-Po and Hershenson, let alone the fact that (iii) what Applicant refer to as GP, for lack of explicit definition, amounts to a mere optimization problem using math rules applying to circuit design, when this problem becomes as complex as in a design of circuit that, *inter alia*, Hershenson's posynomials would be perceived to have mapped such endeavoring. Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims (or even the Specifications in light of the claim) patentably distinguishes them from the references.

(D) Applicants have submitted that for claim 29, neither of the reference teaches 'reduce an algebraic expression to posynomial objective ... equality' (Appl. Rmrks, pg. 16, 5th para); that Shao-Po does not teach signomials, that Hershenson does not create one such, and that Bricker's signomial is mere a solved format. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See In re Keller, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); In re Merck & Co., 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). The rationale of rejection has come from the ground that some approach or application in Shao-Po's mathLab involves circuit complexity; and based on this, combine Shao-Po's equality, inequality in its circuit design math expression with Hershenson's circuit analysis and design via setting of posynomials, both Shao and Hershenson using math polynomial approach. This would render the *posynomial* recited as objective, inequality, or monomial obvious; according to the rationale as set forth in claim 5, but not for the fact of establishing obviousness involving Bricker's negative coefficient. The argument that Hershenson only describes posynomials as opposed to creating them is perceived as incorrect because Hershenson teaches a design problem can be posed as a geometric program (see bottom col. 2) represented via equations of col. 7, or 8 (see Hershenson).

(E) Applicants have submitted that ShaoPo's semidefinite programming and max-det problems can not be extended to GP; and that Examiner appears to have used Official Notice while addressing claim 5 (Appl. Rmrks, pg. 17, middle; pg. 18, top para). This will be referred back to sections B and C from above.

The rejection will stands as set forth.

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Tuan A Vu
Patent Examiner,
Art Unit 2193
September 18, 2006